ECE 595.3

Course Run

ECE595.3: Introduction to Compilers – Optimization

Instructor

- Milind Kulkarni, *Associate Professor of Electrical and Computer Engineering, Purdue University*

Audience

First- or second-year graduate students who have taken ECE 595.1 and ECE 595.2. Students are expected to have programming experience, especially with data structures and recursion.

Course Description

This course covers advanced topics in compiler optimization: dataflow analysis and pointer analysis to perform global optimizations, and both low-level loop transformations such as loop-invariant code motion and high-level loop transformations such as loop tiling. The course also explains how compiler generate code for pointers and arrays. Students will build a compiler that performs a basic pointer analysis.

Course Learning Outcomes

After completing this course, you will be able to:

- Explain how programs handle pointers, and how compilers generate code for them
- Explain how compilers generate code for arrays
- Explain dataflow analysis, including specific instances: liveness and available expressions
- Explain pointer analysis and how to implement it
- Explain loop optimizations and why they are useful
- Extend a compiler to add support for pointers and arrays and implement a pointer analysis

Required Text and Materials

There is no required text for this class. However, students may find the book *Engineering a Compiler*, by Cooper and Torczon (2nd edition, Morgan Kauffman, ISBN: 978-0120884780) helpful.

Gradescope

Gradescope will be used in this course as a method for grading your submitted problem sets

GitHub

Programming assignments will be managed through GitHub classroom

Prerequisites

Comfort with programming, especially data structures and recursion. Experience with object oriented programming will also be beneficial.

Students must have taken ECE 595.1 — Introduction to Compilers – Compiler Basics

Grading

This course will be graded based on the following criteria:

Assessment Type	Description	% of Final Grade
Problem sets	There will be 2 problem sets that will test your knowledge of theoretical aspects of this course	20%
Programming assignments	There will be 2 programming assignments that will walk students through building a compiler that can translate C code into RISC-V assembly	60%
Exam	There will be a final exam summarizing the content of the course	20%

Course Content and Activities

Week	Module	Lessons	Activities & Assignments
1	1.1 Welcome and Introduction	Getting Started/Overview	Lecture Videos Handouts
	1.2 Recap	What have we learned?	Lecture Videos Handouts
	1.3 Pointers	Pointer basics Code generation for pointers Pointer code generation example	Lecture Videos Handouts Quiz: Pointers, l-values and r-values
2	2.1 Arrays	Malloc and Free	Lecture Videos

	2.2 Dataflow Analysis basics	Arrays Desugaring Arrays Addendum: arrays in other languages Dataflow Analysis Constant Propagation Symbolic evaluation Loops and fixpoints	Handouts Quiz: Array and pointer equivalence Lecture Videos Handouts Programming assignment: add pointers and arrays to language
3	3.1 Lattice theory 3.2 More dataflow analysis	Lattice theory Functions on Domains Fixpoint Theorem Lattices More Dataflow Liveness Example Bitvector Dataflow More Bitvector Analysis	Lecture Videos Handouts Lecture Videos Handouts Problem set: Liveness and available expressions
4	4.1 Pointer analysis 4.2 Loop optimization	What is pointer analysis? Flow-sensitive pointer analysis Flow-insensitive pointer analysis Loop optimization Loop invariant code motion Strength reduction	Lecture Videos Handouts Quiz: Pointer analysis Lecture Videos Handouts Programming assignment: Final Project
5	5.1 High-level loop optimization 5.2 Dependences	High level loop optimization Example loop optimizations Dependence Analysis Representing Dependences Loop Parallelization Other Loop Optimizations	Lecture Videos Handouts Problem set: Loop optimization Lecture Videos Handouts

Estimated Effort

- 10-12 hours/week
- 5 weeks total

Languages

Content: English | Videos: English | Transcripts: English

Course Difficulty

- Introductory

Accessibility Support

Purdue University strives to make learning experiences as accessible as possible. If you anticipate or experience physical or academic barriers based on disability, you are welcome to let an instructor know so that we can discuss options. You are also encouraged to contact the Disability Resource Center at: mailto:drc@purdue.edu or by phone: 765-494-1247.

Visit edX's Website Accessibility Policy for information about accessibility on edX.

Course Help

To get help with course content, click the Discussion tab and post a question in "Course Q&A". By commenting in the pinned discussion post, the course team will be able to respond to your question more quickly.

Technical Help

For general questions about using the edX platform, please refer to these resources:

- Technical Documentation: https://docs.edx.org
- Learner Help Center: https://support.edx.org/hc/en-us
- To get help with a technical problem, visit the *Help* link to contact edX Support.

Discussion Guidelines

Please follow the Discussion Guidelines when contributing to discussions in this course. Here are a few of the key points you should remember:

- Do not use offensive language. Present ideas appropriately.
- Be cautious in using Internet language. For example, do not capitalize all letters since this suggests shouting.
- Avoid using vernacular and/or slang language. This could possibly lead to misinterpretation.
- Keep an “open-mind” and be willing to express even your minority opinion.
- Make substantive posts or comments. Avoid comments that do not contribute to the discussion, like "thanks" or "good post."
- Do not hesitate to ask for feedback.
- Be concise and to the point. Give other students the opportunity to join in the discussion.
- Think and edit before you push the “Send” button.

Academic Integrity

Academic integrity is one of the highest values that Purdue University holds. Individuals are encouraged to alert university officials to potential breaches of this value by either emailing or by calling 765-494-8778. While information may be submitted anonymously, the more information that is submitted provides the greatest opportunity for the university to investigate the concern.

The Purdue Honor Pledge

> *"As a boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together - we are Purdue"*

Nondiscrimination Statement

Purdue University is committed to maintaining a community which recognizes and values the inherent worth and dignity of every person; fosters tolerance, sensitivity, understanding, and mutual respect among its members; and encourages each individual to strive to reach his or her own potential. In pursuit of its goal of academic excellence, the University seeks to develop and nurture diversity. The University believes that diversity among its many members strengthens the institution, stimulates creativity, promotes the exchange of ideas, and enriches campus life. Link to Purdue's nondiscrimination policy statement.